

Directed Reading Report

by Muhammad Zubair Khan

Submission date: 04-Dec-2018 02:35PM (UTC+0500)

Submission ID: 1050308992

File name: report_final.pdf (534.76K)

Word count: 5252

Character count: 28632

Directed Reading Report

Muhammad Zubair Khan

`mkzb3@mail.umkc.edu`

December 4, 2018

Abstract

8 Deep neural networks truly have the potential to revolutionize the field of Artificial Intelligence. Main function of Deep Neural Network is to receive a set of inputs, perform progressively complex calculations by passing through set of hidden layers and then use the output to solve the problems. Deep Neural Networks are used for numerous purposes, however this report covers the basic literature in deep learning including the state-of-art research conducted in image classification and segmentation using deep learning. Also, this report target different classification models and architectures applied over various databases using Python libraries. Moreover, deep analysis is performed with the help of key performance metrics and visualizations.

17

Keywords: Neural network, deep learning, convolution network, model, database, architecture

Contents

List of Figures	2
List of Tables	3
1 Introduction	4
1.1 Overview	4
1.1.1 Neural Network's	4
1.1.2 Deep Learning	4
1.2 Literature review	5
2 Data and Tools	7
2.1 Datasets Utilized	7
2.2 Tools Learned and Used	7
2.2.1 Document Writing	7
2.2.2 Code Scripting Tools	8
2.2.3 Code Management Tool	8
3 Models and Results	9
3.1 DiaRetDB1	9
3.1.1 Conv-Net on gray-scale fundus images	9
3.1.2 Conv-net on color fundus images	10
3.1.3 Conv-net on augmented fundus images	11
3.1.4 Transfer learning using VGG16 on color fundus images	12
3.1.5 Transfer learning using VGG19 on color fundus images	13
3.1.6 Fine tuning using VGG16 on color fundus images	14
3.1.7 Fine tuning using VGG19 on color fundus images	15
3.2 Cat-Dog	15
3.2.1 Fully connected deep net on cat-dog images	15
3.2.2 Conv-net on augmented cat-dog images	16
3.3 MNIST	17
3.3.1 Fully connected deep network on mnist repository	17
3.3.2 Conv-net on mnist images	17
3.3.3 Simple autoencoder on mnist images	18

3.3.4	Sparse autoencoder on mnist images	18
3.3.5	deep autoencoder on mnist images	18
3.3.6	Convolutional autoencoder on mnist images	19
3.4	Fashion MNIST	19
3.4.1	Denoising convolutional autoencoder on f-mnist dataset	19
3.5	IMDB	19
3.5.1	Simple RNN on imdb dataset	19
3.5.2	LSTM on imdb repository	20
3.6	Cifar10	21
3.6.1	Conv-net on Cifar10	21
15		
4	Conclusion and Future work	22
4.1	Conclusion	22
4.2	Future work	22
	References	23

List of Figures

3.1	Loss and Accuracy on gray-scale fundus images	10
3.2	Loss and Accuracy on color fundus images	11
3.3	Loss and Accuracy of augmented color fundus images	12
3.4	Loss and Accuracy after applying transfer learning using VGG16 on augmented color fundus images	13
3.5	Loss and Accuracy after applying transfer learning using VGG19 on augmented color fundus images	14
3.6	Loss and Accuracy after applying fine tuning by removing top four layers of VGG16 on augmented color fundus images	14
3.7	Loss and Accuracy after applying fine tuning by removing top four layers of VGG19 on augmented color fundus images	15
3.8	Loss and Accuracy after applying fully connected deep net on cat-dog images	16
3.9	Loss and Accuracy after applying augmented conv-net on cat-dog images	16
3.10	Loss and Accuracy after applying fully connected deep network on mnist dataset	17
3.11	Loss and Accuracy after applying convolution network on mnist dataset	18
3.12	Loss and Accuracy after applying simple rnn on imdb dataset	20
3.13	Loss and Accuracy after applying LSTM on imdb dataset	20
3.14	Loss and Accuracy after applying conv-net on cifar10 dataset	21

List of Tables

1.1	Deep Nets with Datasets	5
3.1	Training and Testing	9
3.2	Classification Report	9
3.3	Training and Testing	10
3.4	Classification Report	10
3.5	Training and Testing	11
3.6	Classification Report	11
3.7	Training and Testing	12
3.8	Classification Report	12
3.9	Training and Testing	13
3.10	Classification Report	13
3.11	Training and Testing	14
3.12	Training and Testing	15
3.13	Training and Testing	16
3.14	Training and Testing	16
3.15	Training and Testing	17
3.16	Training and Testing	17
3.17	Training and Testing loss of simple autoencoder	18
3.18	Training and Testing loss of sparse autoencoder	18
3.19	Training and Testing loss of deep fully connected autoencoder	19
3.20	Training and Testing loss of deep convolutional autoencoder	19
3.21	Training and Testing loss of denoising convolutional autoencoder	19
3.22	Training and Testing	20
3.23	Training and Testing	20
3.24	Training and Testing	21

Chapter 1

Introduction

1.1 Overview

Deep network is state-of-art paradigm designed to allow Artificial Intelligence community to think out of the box. As it is an extension of Neural Nets, it is highly recommended to have enough knowledge of how the Neural Network operates before going deep into Deep Nets.

1.1.1 Neural Network's

8
Neural Network truly have the potential to revolutionize the field of Artificial Intelligence. Main function of Neural Nets is to receive a set of inputs, perform progressively complex calculations and then use the output to solve the problems. Neural networks are used for lot of applications but main goal of this literature is to target classification. Neural Net is highly structured and comes in layers. The first layer is input layer, the final layer is the output layer and all the layers in between are referred to as hidden layers. It mainly operates on Forward Propagation and Backward Propagation.

1.1.2 Deep Learning

Neural Network for complex patterns is unusable, the only practical choice is Deep Network. Deep nets are inspired by the structure of human brain. These are able to break complex patterns down into a series of simpler patterns and contains several number of hidden layers in contrast to shallow nets. The Deep Nets reviewed, are mainly executed for classification over the labeled Databases. The overview of Deep Nets operated against the Datasets in this report can be seen in table: 1.1. Moreover, the code is written in Python3 using Jupyter notebook with Keras library over Tensorflow and accessible on https://github.com/zubairgalbi/DR_Report.

Table 1.1: Deep Nets with Datasets

DIARETDB1	CAT-DOG	CIFAR10	IMDB	MNIST	FASHION-MNIST
Conv-Net	DeepFC	DeepFC	DeepFC	DeepFC	DeepConv-AE
Data-Aug	Conv-Net	Conv-Net	Simple-RNN	Simple-AE	Denoising-AE
TL(VGG16)	Data-Aug	—	LSTM	Sparse-AE	—
TL(VGG19)	—	—	—	DeepFC-AE	—
FT(VGG16)	—	—	—	DeepConv-AE	—
FT(VGG19)	—	—	—	Conv-Net	—

1.2 Literature review

Retinal images plays vital role not only in detection and diagnoses of diseases related to eyes but also helpful in complications of diabetes hypertension and cardiovascular diseases Abràmoff et al. [2010]. There are two common modes for capturing retinal images, which are Fundus Photography and Optical Coherence Tomography. Fundus images provides 2D view whereas, OCT gives cross-sectional view of eye Hassan et al. [2015b]. A survey conducted on Diabetic Retinopathy(DR) in korea in 1998 and it was reported that 44.4/1000 people/year suffers from DR. Later in 2013 this value increased to 56/1000 people/year. This increase mirror the increase of diabetes globally, hence, early detection of DR using automated detection system is useful step Lee and Song [2016]. For analyzing there medical retinal images, Deep learning, specifically deep convolution networks are widely used. Main common tasks performed by deep learning are image classification, object detection, segmentation and object registration Litjens et al. [2017].

Tan et al. [2017a] developed 7-layer conv-net to perform segmentation on 3-channeled fundus images. The output layer contained 4-neurons, exhibiting four categories of segmentation. Average accuracy of 0.9268 is achieved on testing set taken from DRIVE database. Dash and Bhoi [2017] segmented retinal vessels using mean-c local adaptive threshold technique. It shows the mean accuracy of 0.954 and 0.955 for CHASE-DB1 and DRIVE databases respectively. Pratt et al. [2016] has used convnet of 10-convolution layers and 3-fully connected dense layers, trained on 80,000 images of kaggle dataset executed on NVIDIA K40c high-end GPU, achieved the accuracy of 0.75 on 5000 validation images. Zhu et al. [2017] proposed extreme learning machine based supervised method for retinal vessel segmentation using fundus images. Method is tested over DRIVE database and achieved accuracy of 0.9607.

Tan et al. [2017b] used fundus images taken from CLEOPATRA database and performed segmentation using 10-layer convolution neural network on pathological features to discriminate microaneurysms, haemorrhages and exudates. The model achieved the sensitivity of 0.87, 0.71, 0.62, and 0.46 for exudates, dark-lesions, haemorrhages and microaneurysms respectively. Hu et al. [2018] proposed multiscale conv-net with improved loss function for retinal vessel segmentation operated on color fundus images. The model showed better results on tiny thin vessel and vascular edge localization on two public repositories. This method achieved the accuracy of 0.9533 and 0.9632 for DRIVE

and STARE datasets.

Sinhanayothin et al. [2002] proposed a system to analyze color fundus ²⁰ images for features of non-proliferative DR. The method was applied to 30 retinal images for exudates recognition and achieved sensitivity of 0.885 and specificity of 0.997. Its application for haemorrhages and microaneurysms operated on 14 images achieved sensitivity of 0.775 and specificity of 0.887. Prentašić and Lončarić [2016] proposed a method for exudates recognition for early detection of DR using conv-net. Manually segmented database is used for validation. This method achieved F-1 measure of 0.78. Oliveira et al. [2018] proposed novel method that segregates multiscale fully convolution network with stationary wavelet transform to deal with vessel structure in retina. The method is tested on publicly available database CHASE-DB1, STARE and DRIVE with accuracy of 0.9653, 0.9694 and 0.9576 respectively.

Sinhanayothin et al. [1999] used 112 images for localization and segmentation of main features in retinal fundus images. Optic discs are identified using the area with high intensities, blood vessels are found using multi-layer perceptron and foveas were identified using matching correlation. Mean sensitivity and specificity achieved are 0.876 and 0.964 respectively. Early analysis of vascular structures in retinal images plays vital role in curing sight of a person. Hassan et al. [2015a] proposed retinal image segmentation approach to extract blood vessels. Approach used mathematical morphology and K-mean clustering. It is tested on DRIVE dataset and achieved the accuracy of 0.951.

Chapter 2

Data and Tools

2.1 Datasets Utilized

- **DIARETDB1** is multi-class Standard Diabetic Retinopathy Database freely available online for scientific research, contain 208 color fundus images captured with digital fundus camera categorized into 4 classes which are Normal, DR1, DR2, DR3, where DR stands for Diabetic Retinopathy.
- **Cat-Dog** is kaggle platform dataset consist of in-total 25000 3-channeled images, split evenly into 1250 cat and 1250 dog categories, placed in separate folder. .
- **CIFAR10** is small scale public dataset, splited into 50,000 training and 10,000 testing color images designed for classification. Each image is of dimension 32x32.
- **IMDB** is a movie reviews sentiment classification dataset, publicly available online, consist of 50,000 reviews, split evenly into 25000 train and 25000 test sets.
- **MNIST** is a hand written digits database consist of 60,000 training and 10,000 testing subsets. These are grayscale images with dimension of 28x28 and labels ranges from 0 – 9.
- **Fashion-MNIST** is grayscale image dataset contain 60,000 training and 10,000 testing images. The dataset is categorized into 10 fashion categories, designed with images of dimension 28x28.

2.2 Tools Learned and Used

This section covers tools which are essential and learned during the course of semester. These tools are sub-categorized based on their usage.

2.2.1 Document Writing

L^AT_EX is one of the demanding tool majorly use for Article, Report and Book writing. It comes in combination with various add-ones. List of Packages/Tools learned that comes with **L^AT_EX** are enlisted below:

- L^AT_EX (Miktex with TexStudio)
- L^AT_EX (Miktex with TexMaker)
- L^AT_EX with Lyx
- BibTeX management with Mendeley Desktop
- BibTeX management with JabRef
- BibTeX management with Zotero
- Beamer

2.2.2 Code Scripting Tools

Two main development environments are used to write, compile, execute and visualize Computer programs along with numerous packages. These packages are installed using Pip Installer. Description of scripting tools and packages used with each environment is given below:

- Jupyter Notebook with Python 3
 - Numpy
 - Matplotlib
 - Pandas
 - cv2
 - Scikit-learn
 - Keras with TensorFlow
- Pycharm with Python 3
 - Numpy
 - Matplotlib
 - Pandas
 - Scikit-learn

2.2.3 Code Management Tool

In order to outsource, store and manage IPython and Pycharm scripts online on cloud, *Github* repository is used, operated with *Git-Bash* command prompt.

Chapter 3

Models and Results

This chapter is segregated into six subsection. Each subsection reflects peculiar database used, and demonstrates the results attained after executing underlined deep learning strategies.

3.1 DiaRetDB1

Below are the set of Deep Network strategies applied on Fundus images taken from DiaRetDB1 public repository.

3.1.1 Conv-Net on gray-scale fundus images

Architecture consist of 4-convolution, 3-maxpooling and 2-fully connected layers, applied over gray-scale fundus images. The dataset is iterated for 30 epochs segregated in batch size of 32. Number of filters used are 64 with kernel size of 3x3 and pooling filters are of size 2x2. It is observed by looking figure: 3.1 and table: 3.1 that the model is showing positive response in terms of training and testing accuracy and by tuning hyper-parameters, better results could be achieved. Classification report of the model can be seen in table: 3.2.

Table 3.1: Training and Testing

Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
0.8588	0.6154	1.1591	0.4808

Table 3.2: Classification Report

	Precision	Recall	F1-Score	Support
Normal	0.31	0.40	0.35	10
DR1	0.67	0.55	0.60	11
DR2	0.50	0.56	0.53	16
DR3	0.50	0.40	0.44	15
Weighted avg	0.50	0.48	0.48	52

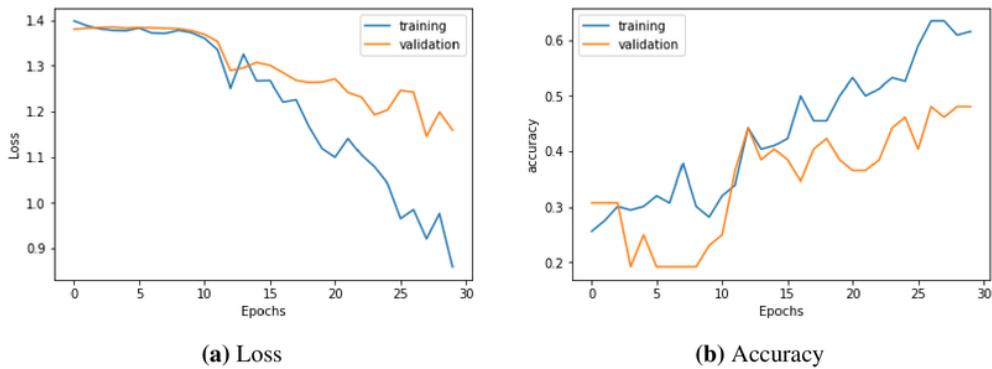


Figure 3.1: Loss and Accuracy on gray-scale fundus images

3.1.2 Conv-net on color fundus images

Architecture consist of 4-convolution, 3-maxpooling and 2-fully connected layers, applied over color fundus images. The dataset is iterated for 30 epochs segregated in batch size of 32. Number of filters used are 64 with kernel size of 3x3 and pooling filters are of size 2x2. It is observed by looking figure: 3.2 and table: 3.3 that the model is showing positive response in terms of training and testing accuracy and by tuning hyper-parameters, better results could be achieved. Classification report of the model can be seen in table: 3.4.

Table 3.3: Training and Testing

Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
0.5568	0.7628	1.3214	0.6923

Table 3.4: Classification Report

	Precision	Recall	F1-Score	Support
Normal	0.53	0.91	0.67	11
DR1	0.90	0.69	0.78	13
DR2	0.73	0.57	0.64	14
DR3	0.75	0.64	0.69	14
Weighted avg	0.73	0.69	0.70	52

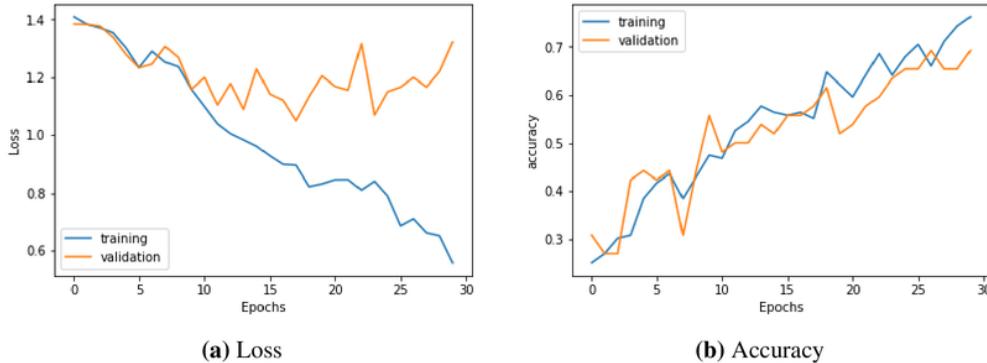


Figure 3.2: Loss and Accuracy on color fundus images

3.1.3 Conv-net on augmented fundus images

Architecture consist of 4-convolution, 3-maxpooling and 2-fully connected layers, applied over color fundus images. The dataset is iterated for 30 epochs segregated in batch size of 32. Number of filters used are 64 with kernel size of 3x3 and pooling filters are of size 2x2. It is observed by looking figure: 3.3 and table: 3.5 that the model is showing positive response in terms of training and testing accuracy and by tuning hyper-parameters, better results could be achieved. Classification report of the model can be seen in table: 3.6.

Table 3.5: Training and Testing

Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
0.9614	0.5495	1.0002	0.4808

Table 3.6: Classification Report

	Precision	Recall	F1-Score	Support
Normal	0.50	0.73	0.59	11
DR1	1.00	0.67	0.80	12
DR2	0.42	0.42	0.42	19
DR3	0.11	0.10	0.11	10
Weighted avg	0.51	0.48	0.48	52

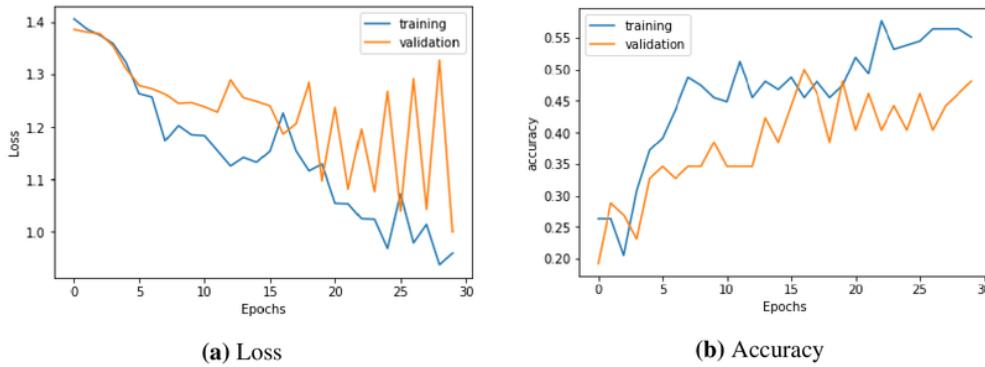


Figure 3.3: Loss and Accuracy of augmented color fundus images

3.1.4 Transfer learning using VGG16 on color fundus images

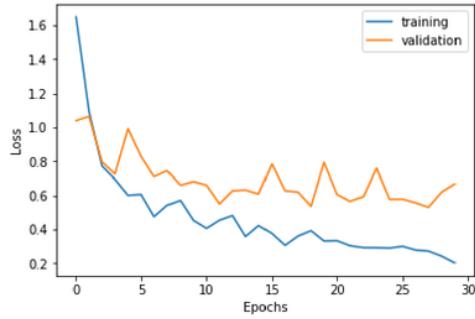
In this subsection classic VGG16 architecture with two fully connected customized layers placed at the top of model along with imagenet weights is used on color fundus images. Batch size of 10 for 30 epochs is considered. Moreover, as number of images are small so data augmentation is performed to increase training and validation samples in order to reduce chances of over-fitting. The result given in table: 3.7 and figure: 3.4 shows that VGG16 with imagenet weights performed considerably well on generalized dataset of fundus images. The classification report is given in table: 3.8.

Table 3.7: Training and Testing

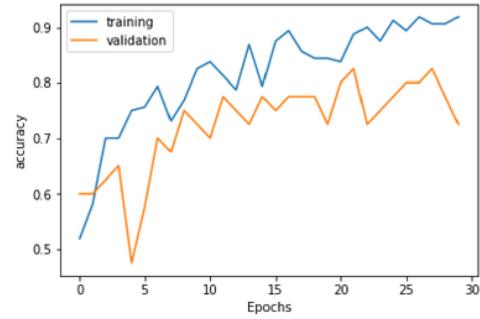
Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
0.2025	0.9187	0.6669	0.7250

Table 3.8: Classification Report

	Precision	Recall	F1-Score	Support
Normal	0.40	0.29	0.33	7
DR1	0.94	0.74	0.83	23
DR2	0.59	1.00	0.74	10
Weighted avg	0.76	0.72	0.72	40



(a) Loss



(b) Accuracy

Figure 3.4: Loss and Accuracy after applying transfer learning using VGG16 on augmented color fundus images

3.1.5 Transfer learning using VGG19 on color fundus images

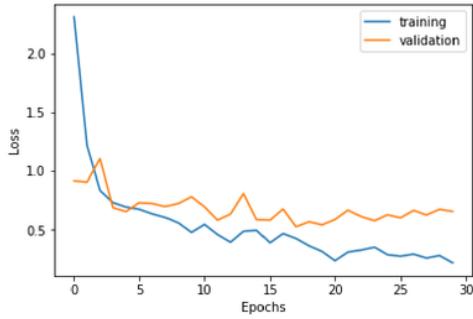
In this subsection classic VGG19 architecture with two fully connected customized layers placed at the top of model along with imagenet weights is used on color fundus images. Batch size of 10 for 30 epochs is considered. Moreover, as number of images are small so data augmentation is performed to increase training and validation samples in order to reduce chances of over-fitting. The result given in table: 3.9 and figure: 3.5 shows that VGG19 with imagenet weights performed considerably well on generalized dataset of fundus images. The classification report is given in table: 3.10.

Table 3.9: Training and Testing

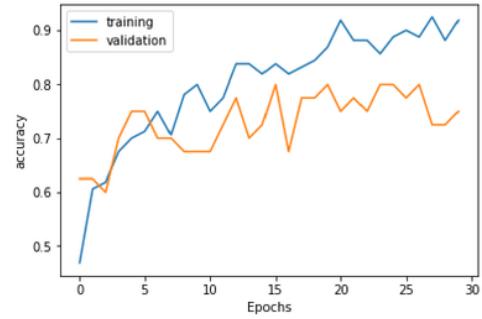
Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
0.2166	0.9187	0.6566	0.7500

Table 3.10: Classification Report

	Precision	Recall	F1-Score	Support
Normal	0.60	0.43	0.50	7
DR1	0.85	0.74	0.79	23
DR2	0.67	1.00	0.80	10
Weighted avg	0.76	0.75	0.74	40



(a) Loss



(b) Accuracy

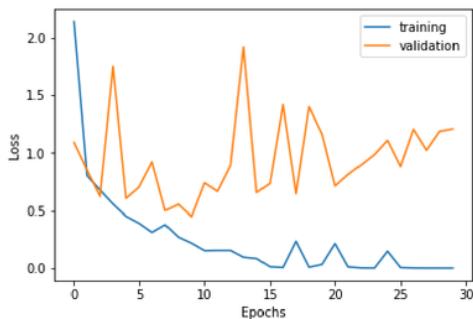
Figure 3.5: Loss and Accuracy after applying transfer learning using VGG19 on augmented color fundus images

3.1.6 Fine tuning using VGG16 on color fundus images

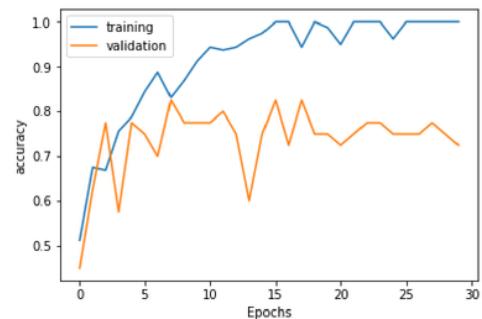
In this subsection classic VGG16 architecture after eliminating top 4 layers is considered. After this two fully connected customized layers placed at the top of model along with imagenet weights. Batch size of 10 for 30 epochs is considered. Moreover, data augmentation is performed as pre-processing step before feeding on VGG16 network. The result given in table: 3.11 and figure: 3.6 shows that VGG16 with imagenet weights performed considerably well on generalized dataset of fundus images.

Table 3.11: Training and Testing

Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
$3.2858e - 05$	1.0000	1.2077	0.7250



(a) Loss



(b) Accuracy

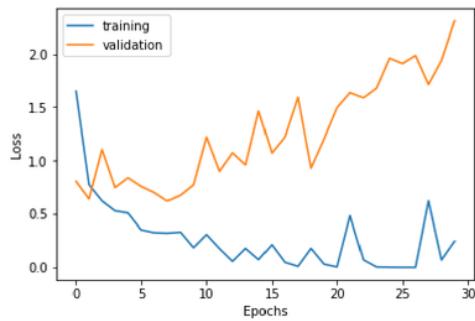
Figure 3.6: Loss and Accuracy after applying fine tuning by removing top four layers of VGG16 on augmented color fundus images

3.1.7 Fine tuning using VGG19 on color fundus images

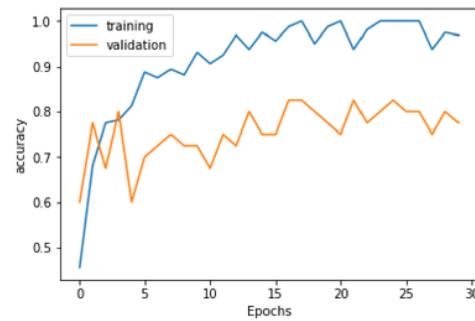
In this subsection classic VGG19 architecture after eliminating top 4 layers is considered. After this two fully connected customized layers placed at the top of model along with imagenet weights. Batch size of 10 for 30 epochs is considered. Moreover, data augmentation is performed as pre-processing step before feeding on VGG16 network. The result given in table: 3.12 and figure: 3.7 shows that VGG19 with imagenet weights performed considerably well on generalized dataset of fundus images.

Table 3.12: Training and Testing

1	Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
	0.2444	0.9687	2.3069	0.7750



(a) Loss



(b) Accuracy

Figure 3.7: Loss and Accuracy after applying fine tuning by removing top four layers of VGG19 on augmented color fundus images

3.2 Cat-Dog

Deep learning models applied on Cat-Dog data repository is given in following subsections.

3.2.1 Fully connected deep net on cat-dog images

Architecture consist of 4-deep layers, applied over colored images. The dataset is iterated for 10 epochs segregated in batch size of 64. Number of filters used are 32 with kernel size of 3x3 and pooling filters are of size 2x2. It is observed by looking figure: 3.8 and table: 3.13 that the model is showing positive response in terms of training and testing accuracy and by tuning hyper-parameters, better results could be achieved.

Table 3.13: Training and Testing

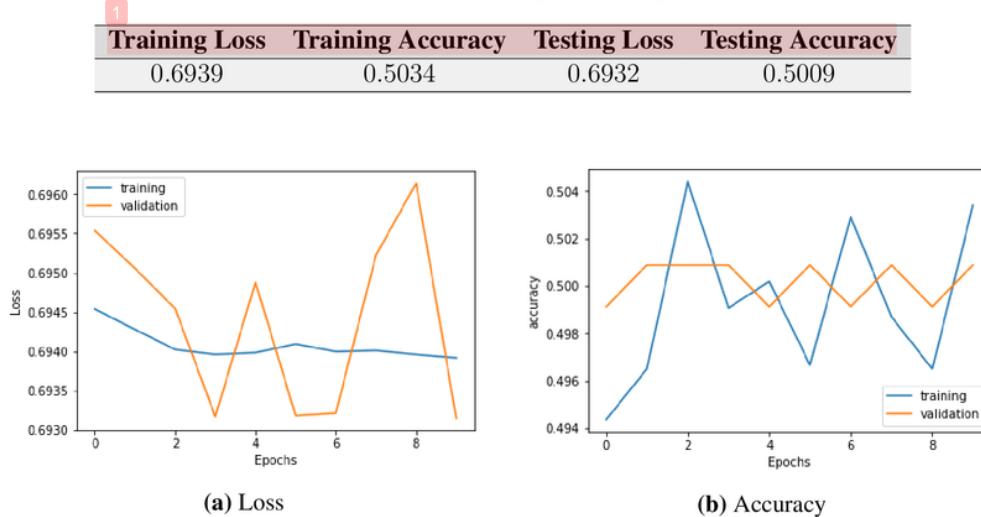


Figure 3.8: Loss and Accuracy after applying fully connected deep net on cat-dog images

3.2.2 Conv-net on augmented cat-dog images

Architecture consist of 2-conv-layers,1-maxpool layer and 2-finally connected dense layers, applied over colored images. The dataset is iterated for 10 epochs segregated in batch size of 64. Number of filters used are 32 with kernel size of 3x3 and pooling filters are of size 2x2. It is observed by looking figure: 3.9 and table: 3.14 that the model is showing positive response in terms of training and testing accuracy and by tuning hyper-parameters, better results could be achieved.

Table 3.14: Training and Testing

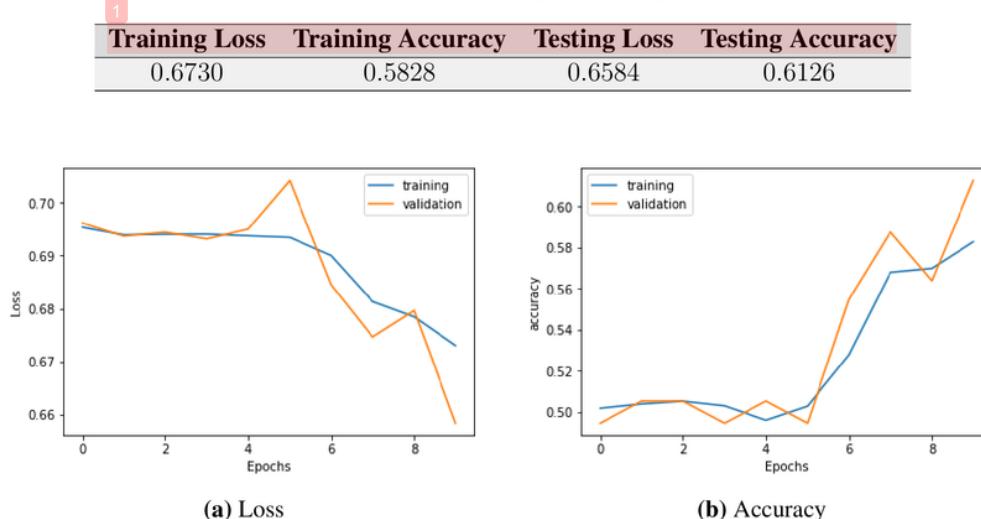


Figure 3.9: Loss and Accuracy after applying augmented conv-net on cat-dog images

3.3 MNIST

Deep learning models applied on MNIST data repository is given in following subsections.

3.3.1 Fully connected deep network on mnist repository

The architecture consist of 3-dense layers operated with batch size of 128 for 10 epochs. It is observed by looking figure: 3.10 and table: 3.15 that the model is showing positive response in terms of training and testing accuracy and by tuning hyper-parameters, better results could be achieved.

Table 3.15: Training and Testing

Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
0.6807	0.8497	0.6627	0.8509

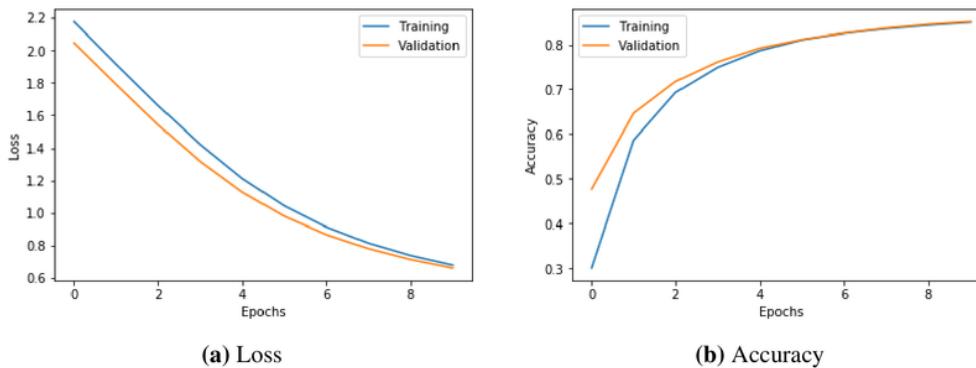


Figure 3.10: Loss and Accuracy after applying fully connected deep network on mnist dataset

3.3.2 Conv-net on mnist images

Architecture consist of 2-conv-layers,1-maxpool layer and 2-finally connected dense layers, applied over colored images. The dataset is iterated for 10 epochs segregated in batch size of 64. Number of filters used are 32 with kernel size of 3x3 and pooling filters are of size 2x2. It is observed by looking figure: 3.11 and table: 3.16 that the model is showing positive response in terms of training and testing accuracy and by tuning hyper-parameters, better results could be achieved.

Table 3.16: Training and Testing

Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
0.2205	0.9451	0.1550	0.9561

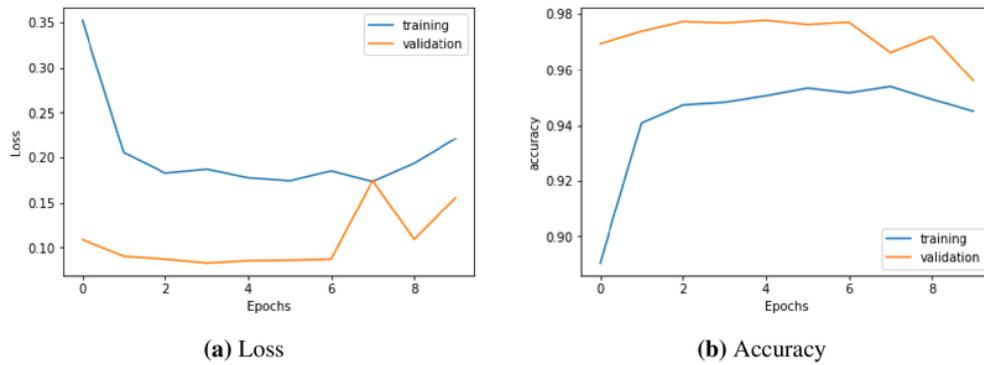


Figure 3.11: Loss and Accuracy after applying convolution network on mnist dataset

3.3.3 Simple autoencoder on mnist images

Model consist of 2 dense layers, one for encoding and other for decoding executed for 50 epochs. Input shape taken is 784 whereas, encoding dimension is 32. Training and validation losses are mentioned in table: 3.17.

Table 3.17: Training and Testing loss of simple autoencoder

Training Loss	Testing Loss
0.1030	0.1013

3.3.4 Sparse autoencoder on mnist images

Model consist of 2 dense layers, one for encoding and other for decoding executed for 50 epochs. Input shape taken is 784 whereas, encoding dimension is 32. Training and validation losses shown in table: 3.18 depicts that simple autoencoders performs better than sparse autoencoders on mnist dataset.

Table 3.18: Training and Testing loss of sparse autoencoder

Training Loss	Testing Loss
0.2963	0.2960

3.3.5 deep autoencoder on mnist images

Model consist of 6 dense layers in total, three are encoding layers and three are decoding, executed for 50 epochs. Input shape taken is 784 whereas, final encoding dimension is 32. Training and validation losses shown in table: 3.19 depicts that deep autoencoders performs better than sparse autoencoders but shows poor performance compare to simple autoencoders on mnist dataset.

Table 3.19: Training and Testing loss of deep fully connected autoencoder

Training Loss	Testing Loss
0.1109	0.1124

3.3.6 Convolutional autoencoder on mnist images

Model consist of 7-conv-layers,3-maxpooling layers and 3-upsampling layers. Input shape taken is 784. Training and validation losses shown in table: 3.20 depicts that deep autoencoders outperforms all other autoencoders on mnist dataset.

Table 3.20: Training and Testing loss of deep convolutional autoencoder

Training Loss	Testing Loss
0.0980	0.0962

3.4 Fashion MNIST

Deep learning strategies applied on Fashion-MNIST data repository is given below.

3.4.1 Denoising convolutional autoencoder on f-mnist dataset

The architecture consist of 7-convolution layers, 3-Maxpool layers and 3-upsampling layers operated over fashion-mnist dataset. The performance of denoising convolutional autoencoder can be seen in table: 3.21.

Table 3.21: Training and Testing loss of denoising convolutional autoencoder

Training Loss	Testing Loss
0.3088	0.3437

3.5 IMDB

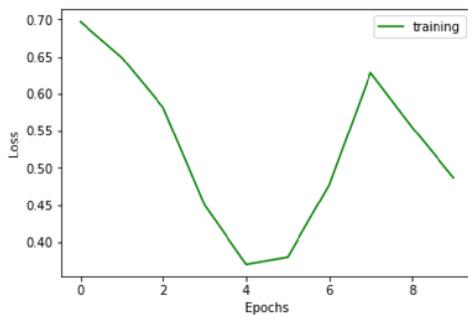
IMDB is sequential data repository publicly available for researcher. Fortunately deep learning is rich in dealing with sequential data. The deep strategies applied over imdb dataset is given down below.

3.5.1 Simple RNN on imdb dataset

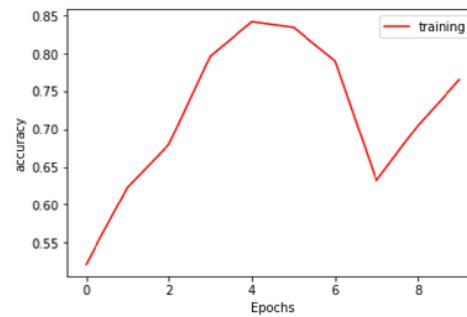
For executing simple rnn model, 5000 top words with maximum review length of 500 is considered. Sequence padding is performed as preprocessing step before feeding data in model. The model shows reasonable performance on underlined data set which is depicted in figure: 3.12 and table: 3.22.

Table 3.22: Training and Testing

1	Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
	0.4865	0.7653	0.6700	0.6100



(a) Loss



(b) Accuracy

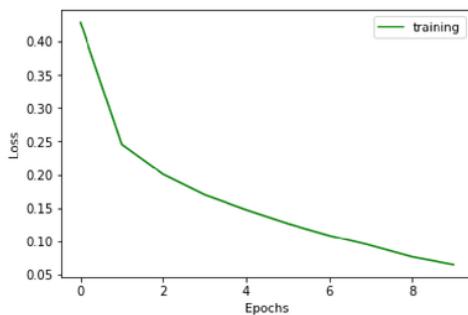
Figure 3.12: Loss and Accuracy after applying simple rnn on imdb dataset

3.5.2 LSTM on imdb repository

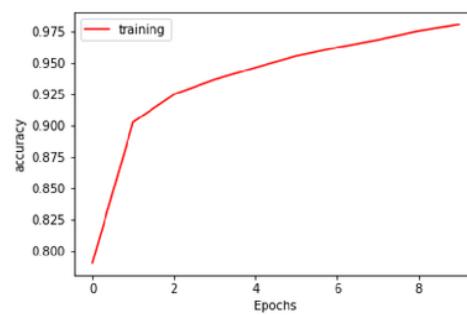
LSTM model is executed on imbd repository for 10 epochs by considering 5000 top words with maximum review length of 500. Sequence padding is performed as preprocessing step before feeding data in model. The model shows reasonable performance on underlined data set which is depicted in figure: 3.13 and table: 3.23.

Table 3.23: Training and Testing

1	Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
	0.0648	0.9807	0.4254	0.8633



(a) Loss



(b) Accuracy

Figure 3.13: Loss and Accuracy after applying LSTM on imbd dataset

3.6 Cifar10

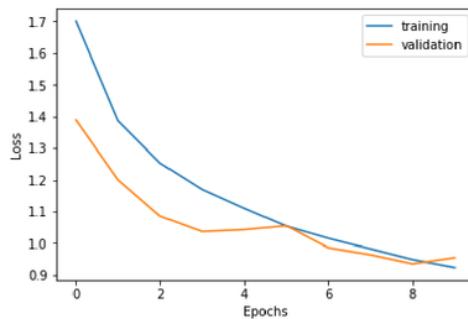
Deep learning model applied on cifar10 data repository is given below.

3.6.1 Conv-net on Cifar10

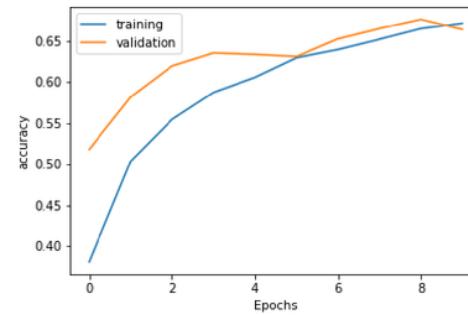
Cifar10 is executed on model with 2-convolution layers,single Maxpool layer and 2-fully connected dense layers.The model is executed for 10 epochs with batch-size of 64 operated on 32 filter of kernel size 3x3 and pool size of 2x2. The model shows reasonable performance mentioned in table: 3.24 even for 10 epochs and shown in figure: 3.14.

Table 3.24: Training and Testing

Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
0.9235	0.6712	0.9545	0.6637



(a) Loss



(b) Accuracy

Figure 3.14: Loss and Accuracy after applying conv-net on cifar10 dataset

Chapter 4

Conclusion and Future work

4.1 Conclusion

Deep learning is widely used in medical imaging, specially after the advancement in imaging technology. One of the core area where researchers are showing interest is ophthalmology. Retinal images contain features which not only help in diagnosis of eye disease but also problem related to circulation and cardiovascular. Two common technologies are used for capturing retinal images, one is fundus photography and other is optical coherence tomography. This report covers state-of-art research conducted on Diabetic Retinopathy using fundus images and deep networks. Also, multiple deep learning networks are applied on six different data repositories and their performance is analyzed and measured.

4.2 Future work

The future work focuses on studying and executing various published deep neural architectures. Also, Image segmentation using deep neural networks would be the key area of interest.

References

- 13 Michael D Abràmoff, Mona K Garvin, and Milan Sonka. Retinal imaging and image analysis. *IEEE reviews in biomedical engineering*, 3:169–208, 2010.
- 12 Jyotiprava Dash and Nilamani Bhoi. A thresholding based technique to extract retinal blood vessels from fundus images. *Future Computing and Informatics Journal*, 2(2):103–109, 2017.
- 24 Gehad Hassan, Nashwa El-Bendary, Aboul Ella Hassanien, Ali Fahmy, Vaclav Snasel, et al. Retinal blood vessel segmentation approach based on mathematical morphology. *Procedia Computer Science*, 65:612–622, 2015a.
- 3 Taimur Hassan, M Usman Akram, Bilal Hassan, Ammara Nasim, and Shafaat Ahmed Bazaz. Review of oct and fundus images for detection of macular edema. In *Imaging Systems and Techniques (IST), 2015 IEEE International Conference on*, pages 1–4. IEEE, 2015b.
- 7 Kai Hu, Zhenzhen Zhang, Xiaorui Niu, Yuan Zhang, Chunhong Cao, Fen Xiao, and Xieping Gao. Retinal vessel segmentation of color fundus images using multiscale convolutional neural network with an improved cross-entropy loss function. *Neurocomputing*, 2018.
- 16 16 Jae Hyuck Lee and Su Jeong Song. Current challenges in diabetic retinopathy: are we really doing better? *Endocrinology and Metabolism*, 31(2):254–257, 2016.
- 8 Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42: 60–88, 2017.
- Américo Filipe Moreira Oliveira, Sérgio Rafael Mano Pereira, and Carlos Alberto Batista Silva. Retinal vessel segmentation based on fully convolutional neural networks. *Expert Systems with Applications*, 2018.
- 5 Harry Pratt, Frans Coenen, Deborah M Broadbent, Simon P Harding, and Yalin Zheng. Convolutional neural networks for diabetic retinopathy. *Procedia Computer Science*, 90:200–205, 2016.
- Pavle Prentašić and Sven Lončarić. Detection of exudates in fundus photographs using deep neural networks and anatomical landmark detection fusion. *Computer methods and programs in biomedicine*, 137:281–292, 2016.

4

Chanjira Sinthanayothin, James F Boyce, Helen L Cook, and Thomas H Williamson. Automated localisation of the optic disc, fovea, and retinal blood vessels from digital colour fundus images. *British Journal of Ophthalmology*, 83(8):902–910, 1999.

11

Chanjira Sinthanayothin, James F Boyce, Tom H Williamson, Helen L Cook, Evelyn Mensah, Shantanu Lal, and David Usher. Automated detection of diabetic retinopathy on digital fundus images. *Diabetic medicine*, 19(2):105–112, 2002.

2

Jen Hong Tan, U Rajendra Acharya, Sulatha V Bhandary, Kuang Chua Chua, and Sobha Sivaprasad. Segmentation of optic disc, fovea and retinal vasculature using a single convolutional neural network. *Journal of Computational Science*, 20:70–79, 2017a.

6

Jen Hong Tan, Hamido Fujita, Sobha Sivaprasad, Sulatha V Bhandary, A Krishna Rao, Kuang Chua Chua, and U Rajendra Acharya. Automated segmentation of exudates, haemorrhages, microaneurysms using single convolutional neural network. *Information sciences*, 420:66–76, 2017b.

10

Chengzhang Zhu, Beiji Zou, Rongchang Zhao, Jinkai Cui, Xuanchu Duan, Zailiang Chen, and Yixiong Liang. Retinal vessel segmentation in colour fundus images using extreme learning machine. *Computerized Medical Imaging and Graphics*, 55:68–77, 2017.

Directed Reading Report

ORIGINALITY REPORT



PRIMARY SOURCES

- | | | |
|---|---|------------|
| 1 | mdpi.com
Internet Source | 2% |
| 2 | export.arxiv.org
Internet Source | 2% |
| 3 | Samra Naz, M. Usman Akram, Shoab A. Khan.
"Automated segmentation of retinal layers
from OCT images using structure tensor and
kernel regression+GTDP approach", 2017 1st
International Conference on Next Generation
Computing Applications (NextComp), 2017
Publication | 1 % |
| 4 | Submitted to University of KwaZulu-Natal
Student Paper | 1 % |
| 5 | Xiaogang Li, Tiantian Pang, Biao Xiong,
Weixiang Liu, Ping Liang, Tianfu Wang.
"Convolutional neural networks based transfer
learning for diabetic retinopathy fundus image
classification", 2017 10th International
Congress on Image and Signal Processing,
BioMedical Engineering and Informatics (CISP- | 1 % |

- | | | |
|----|---|------|
| 6 | onlinelibrary.wiley.com
Internet Source | 1 % |
| 7 | dblp.uni-trier.de
Internet Source | 1 % |
| 8 | opensourceforu.com
Internet Source | 1 % |
| 9 | Submitted to Queen Mary and Westfield College
Student Paper | 1 % |
| 10 | Toufique A. Soomro, Manoranjan Paul, Junbin Gao, Lihong Zheng. "Retinal blood vessel extraction method based on basic filtering schemes", 2017 IEEE International Conference on Image Processing (ICIP), 2017
Publication | 1 % |
| 11 | Submitted to Amrita Vishwa Vidyapeetham
Student Paper | 1 % |
| 12 | Submitted to Higher Education Commission Pakistan
Student Paper | 1 % |
| 13 | lutpub.lut.fi
Internet Source | <1 % |
| 14 | Submitted to College of Engineering | |

15

fedetd.mis.nsysu.edu.tw

Internet Source

<1 %

16

Jae Hyuck Lee, Su Jeong Song. "Current Challenges in Diabetic Retinopathy: Are We Really Doing Better?", *Endocrinology and Metabolism*, 2016

Publication

<1 %

17

"Classification in BioApps", Springer Nature, 2018

Publication

<1 %

18

Thamer Mitib Al Sariera, Lalitha Rangarajan. "Extraction of Blood Vessels in Retina", *Journal of Information Technology Research*, 2018

Publication

<1 %

19

link.lib.umanitoba.ca

Internet Source

<1 %

20

C. Sinthanayothin. "Automated detection of diabetic retinopathy on digital fundus images", *Diabetic Medicine*, 2/2002

Publication

<1 %

21

Jen Hong Tan, Hamido Fujita, Sobha Sivaprasad, Sulatha V. Bhandary, A. Krishna Rao, Kuang Chua Chua, U. Rajendra Acharya. "Automated segmentation of exudates,

<1 %

haemorrhages, microaneurysms using single convolutional neural network", *Information Sciences*, 2017

Publication

22

Américo Filipe Moreira Oliveira, Sérgio Rafael Mano Pereira, Carlos Alberto Batista Silva. "Retinal Vessel Segmentation based on Fully Convolutional Neural Networks", *Expert Systems with Applications*, 2018

<1 %

Publication

23

Ammar Gharaibeh, Mohammad A. Salahuddin, Sayed Jared Hussini, Abdallah Khreishah, Issa Khalil, Mohsen Guizani, Ala Al-Fuqaha. "Smart Cities: A Survey on Data Management, Security, and Enabling Technologies", *IEEE Communications Surveys & Tutorials*, 2017

<1 %

Publication

24

Submitted to University College London

<1 %

Student Paper

25

Chengzhang Zhu, Beiji Zou, Rongchang Zhao, Jinkai Cui, Xuanchu Duan, Zailiang Chen, Yixiong Liang. "Retinal vessel segmentation in colour fundus images using Extreme Learning Machine", *Computerized Medical Imaging and Graphics*, 2017

<1 %

Publication

26

Nogol Memari, Abd Rahman Ramli, M. Iqbal

Bin Saripan, Syamsiah Mashohor, Mehrdad Moghbel. "Supervised retinal vessel segmentation from color fundus images based on matched filtering and AdaBoost classifier", PLOS ONE, 2017

Publication

<1 %

27 Jyotiprava Dash, Nilamani Bhoi. "An Unsupervised Approach for Extraction of Blood Vessels from Fundus Images", Journal of Digital Imaging, 2018

Publication

<1 %

28 sujitpal.blogspot.tw
Internet Source

<1 %

29 www.ijeat.org
Internet Source

<1 %

30 amslaurea.unibo.it
Internet Source

<1 %

31 Kai Hu, Zhenzhen Zhang, Xiaorui Niu, Yuan Zhang, Chunhong Cao, Fen Xiao, Xieping Gao. "Retinal vessel segmentation of color fundus images using multiscale convolutional neural network with an improved cross-entropy loss function", Neurocomputing, 2018

Publication

<1 %

32 Zou, Beiji, Jinkai Cui, Chengzhang Zhu, Yao Xiang, and Hui Wu. "An Ensemble Retinal

<1 %

Vessel Segmentation Based on Supervised Learning in Fundus Images", Chinese Journal of Electronics, 2016.

Publication

Exclude quotes Off

Exclude bibliography Off

Exclude matches Off